Please amend the above-identified patent application as follows:

In the claims:

1. (currently amended) A computing system for decoding a Reed-Solomon-encoded string of data, the computing system comprising a processor circuit operable to:

store a portion of a first ~~receive a~~ Reed-Solomon code word, the portion being less than the entire first code word;

store a portion of a second Reed-Solomon code word, the portion being less than the entire second code word; and

while storing the portion of the second code word, decodinge the portion of the first code word and no other portion of the first ~~the~~ code word ~~in software with a constant or approximately constant level of processing~~.

2. (currently amended) The computing system of claim 1 wherein:

storing the portions of the first and second Reed-Solomon code words ~~receiving the code word~~ comprises storing each of the portions ~~receiving the code word~~ in a respective time having a duration *t*; and

decoding the portion of the first code word comprises decoding the portion ~~code word~~ in ~~a number of steps each having a~~the duration ~~of~~ *t*.

3 -4. Canceled.

5. (currently amended) A computing system for decoding a Reed-Solomon-encoded string of data, the computing system comprising a processor circuit operable to:

receive *m* portions of a first Reed-Solomon code word, *m* being greater than one;

receive *m* portions of a second Reed-Solomon code word after receiving the first code word; ~~and~~

while receiving a first portion of the second code word, decoding a first portion of the first code word and no other portion of the first code word; and

while receiving a second portion of the second code word, decoding the second portion of the first code word and no other portion of the first code word~~the first decoding step of the first code word with a constant or approximately constant level of processing~~.

6. (currently amended) The computing system of claim 5 wherein the processor circuit is further operable to:

receive~~ing~~ the first and second portions of the second code word ~~comprises receiving the second code word~~ during ~~a~~ respective first and second time period~~s~~ each having a same duration ~~*t*~~; and

decode~~ing~~ the first and second portions of the first code word during the first and second time periods respectively~~comprises performing a first decoding step of the first code word during the time period~~.

7. (currently amended) The computing system of claim 5 wherein the processor circuit is ~~,~~ further operable to ~~comprising~~:

receive *m* portions of a third Reed-Solomon code word;

while receiving a first portion of the third code word, decoding a first portion of the second code word and no other portion of the second code word; and

while receiving a second portion of the third code word, decoding the second portion of the second code word and no other portion of the second code word.

~~wherein receiving the second code word comprises receiving the second code word during a first time period having a duration *t*;~~

~~wherein decoding the first code word comprises performing a first decoding step on the first code word during the first time period;~~

~~receiving a third code word during a second time period having the duration *t*;~~

~~performing a second decoding step on the first code word during the second time period; and~~

~~performing the first decoding step on the second code word during the second time period.~~

8. (currently amended) The computing system of claim 5 wherein:

the first and second Reed-Solomon code words each comprise ~~a number~~ *n* symbols~~of equal-length blocks~~; and

n/m equals an integer that is greater than one

~~receiving the first and second code words comprises receiving each of the blocks in *n* sequential time periods each having a duration *t/n*; and~~

~~decoding the first code word comprises decoding the first code word in *n* sequential sub-steps each having a duration *t/n*.~~

9. (currently amended) A computing system for decoding a Reed-Solomon-encoded string of data, the computing system comprising a processor circuit operable to:

receive *m* portions of each of five Reed-Solomon code words~~, each~~ during respective ~~a~~ time periods each ~~-~~having a duration ~~*t*~~, ~~and~~ each code word comprising ~~a number *n* of equal-length blocks~~ symbols such that *n/m* equals an integer that is greater than one;

while receiving a first portion of the second code word, decoding a first portion of the first code word according to a first algorithm and decoding no other portion of the first code word~~the first step of the first code word in *n* sequential sub-steps each having a duration *t/n*~~;

while receiving a first portion of the third code word, decoding a first portion of the second code word according to the first algorithm, decoding the first portion of the first code word according to a second algorithm, and decoding no other portions of the first and second code words~~the second step of the first code word in *n* sequential sub-steps each having a duration *t/n* and decoding the first step of the second code word in *n* sequential sub-steps each having a duration *t/n*~~;

while receiving a first portion of the fourth code word, decoding a first portion of

the third code word according to the first algorithm, decoding the first portion of the second code word according to the second algorithm, decoding the first portion of the first code word according to a third algorithm, and decoding no other portions of the first, second, and third code words~~decoding the third step of the first code word in *n* sequential sub-steps each having a duration *t/n* and decoding the second step of the second code word in *n* sequential sub-steps each having a duration *t/n* and decoding the first step of the third code word in *n* sequential sub-steps each having a duration *t/n*~~; and

while receiving a first portion of the fifth code word, decoding a first portion of the fourth code word according to the first algorithm, decoding the first portion of the third code word according to the second algorithm, decoding the first portion of the second code word according to the third algorithm, decoding the first portion of the first code word according to a fourth algorithm, and decoding no other portions of the first, second, third, and fourth code words~~while receiving the fifth code word, decoding the fourth step of the first code word in *n* sequential sub-steps each having a duration *t/n* and decoding the third step of the second code word in *n* sequential sub-steps each having a duration *t/n* and decoding the second step of the third code word in *n* sequential sub-steps each having a duration *t/n* and decoding the first step of the fourth code word in *n* sequential sub-steps each having a duration *t/n*.~~

10. (currently amended) A computing system for decoding a Reed-Solomon-encoded string of data, the computing system comprising a processor circuit operable to:

~~1.~~ sequentially receive *m* portions of the Reed-Solomon-encoded string of data in *T*~~t~~ seconds, the string of data including *n* symbols, *n/m* equaling a first integer that is greater than one;

~~2.~~ sequentially calculate *m* ~~a~~ respective partial syndromes for the *m* portions of the string, the processor circuit operable to calculate each of the *m* partial syndromes in *T/m* ~~in said *t*~~ seconds;

3. from the *m* partial syndromes, sequentially calculate the coefficients of *m*a respectiven sets of error locator polynomials, the processor circuit operable to calculate each set of coefficients in *T/m*said *t* seconds;

4. sequentially determine *m* respective sets of the roots for the sets of theof an error locator polynomials, the processor circuit operable to determine each set of roots in said *t**T/m* seconds; and

for each of the *m* sets of roots, sequentially determine the magnitude of an respective error in *T/m* seconds; and

sequentially correcting each of the errors in *T/m* seconds.

11. (currently amended) The computing system of claim 10 wherein:

said *n* symbols each comprise b bitssaid Reed-Solomon-encoded string of data comprises:

a first number of data symbols each including second number of bits; and

k of said n symbols comprise data symbolsa third number of parity symbols each including the second number of bits;

n-k of said symbols comprise parity symbols.;

(n-k)/m equals a second integer; and

k/m equals a third integer.

12. Canceled.

13. (original) The computing system of claim 10 wherein the processor circuit executes a Berlekamp-Massey Algorithm to calculate the coefficients of the error locator polynomial.

14. (original) The computing system of claim 10 wherein the processor circuit executes a Chien Search to determine the roots of the error locator polynomial.

15. (original) The computing system of claim 10 wherein the processor circuit

executes a Forney Algorithm to determine the magnitude of the errors in the received digital code word.

16 - 18. Canceled.

19. (currently amended) A method of operating a computing system with a Reed-Solomon decoding application comprising the steps of:

receiving a portion of a first Reed-Solomon code word, the portion being less than the whole first code word;

receiving a portion of a second Reed-Solomon code word, the portion being less than the whole second code word; and

while receiving the portion of the second code word, decoding the portion of the first code word and no other portion of the first code word~~in software with a constant or approximately constant level of processing~~.

20. (currently amended) The method of claim 19 wherein the portion of the first Reed-Solomon code word is the same length as the portion of the second Reed-Solomon code word~~:~~

~~receiving the code word comprises receiving the code word in a time having a duration *t*; and~~

~~decoding the code word comprises decoding the code word in a number of steps each having a duration of *t*.~~

21. (currently amended) The method of claim 19 wherein:

receiving the portion of the second code word comprises receiving the portion of the second code word in a time ~~having a duration *t*~~*T*; and

decoding the portion of the first code word comprises decoding the portion of the first code word in the time *T*~~code word in a number of sequential steps each having a duration of *t*~~.

22. (currently amended) The method of claim 19 wherein:

the first and second code words each comprises ~~a number~~ *m~~n~~* portions of equal-length ~~blocks;~~

~~receiving the code word comprises receiving each of the blocks in a time having a duration *t/n*; and~~

~~decoding the code word comprises decoding the code word in a number of sequential steps each having *n* sub-steps of a duration *t/n*~~.

23. (currently amended) A method of operating a computing system with a Reed-Solomon decoding application comprising the steps of:

receiving *m* portions of a first Reed-Solomon code word, *m* being greater than one ~~a first Reed-Solomon code word~~;

receiving *m* portions of a second Reed-Solomon code word after receiving the first code word ~~a second Reed-Solomon code word after receiving the first code word~~; and

while receiving a first portion of the second code word, decoding a first portion of the first code word and no other portion of the first code word; and

while receiving a second portion of the second code word, decoding the second portion of the first code word and no other portion of the first code word. ~~while receiving the second code word, decoding the first decoding step of the first code word with a constant or approximately constant level of processing~~.

24. (currently amended) The method of claim 23 wherein:

receiving the *m* portions of the first and second code words comprises receiving each of the *m* portions during a respective period of time T ~~the second code word during a time period having a duration *t*~~; and

decoding each of the first and second portions of the first code word during the same respective periods that the first and second portions of the second code word are received ~~comprises performing a first decoding step of the first code word during the time period~~.

25. Cancel.

26. (currently amended) The method of claim 23 wherein:

the first and second code words each comprise ~~a number~~ *n* ~~of~~ equal-length symbols~~blocks~~;

~~receiving the first and second code words comprises receiving each of the blocks in *n* sequential time periods each having a duration *t/n*~~; and

n/m equals an integer greater than one~~decoding the first code word comprises decoding the first code word in *n* sequential sub-steps each having a duration *t/n*~~.

27. (currently amended) A method of operating a computing system with a Reed-Solomon decoding application comprising the steps of:

receiving *m* portions of each of five Reed-Solomon code words during respective time periods each having a duration, each code word comprising *n* symbols such that *n/m* equals an integer that is greater than one;

while receiving a first portion of the second code word, decoding a first portion of the first code word according to a first algorithm and decoding no other portion of the first code word;

while receiving a first portion of the third code word, decoding a first portion of the second code word according to the first algorithm, decoding the first portion of the first code word according to a second algorithm, and decoding no other portions of the first and second code words;

while receiving a first portion of the fourth code word, decoding a first portion of the third code word according to the first algorithm, decoding the first portion of the second code word according to the second algorithm, decoding the first portion of the first code word according to a third algorithm, and decoding no other portions of the first, second, and third code words; and

while receiving a first portion of the fifth code word, decoding a first portion of the fourth code word according to the first algorithm, decoding the first portion of the third

code word according to the second algorithm, decoding the first portion of the second code word according to the third algorithm, decoding the first portion of the first code word according to a fourth algorithm, and decoding no other portions of the first, second, third, and fourth code words~~receiving five Reed-Solomon code words, each during a time period having a duration *t*, and each comprising a number *n* of equal-length blocks;~~

~~while receiving the second code word, decoding the first step of the first code word in *n* sequential sub-steps each having a duration *t/n*;~~

~~while receiving the third code word, decoding the second step of the first code word in *n* sequential sub-steps each having a duration *t/n* and decoding the first step of the second code word in *n* sequential sub-steps each having a duration *t/n*;~~

~~while receiving the fourth code word, decoding the third step of the first code word in *n* sequential sub-steps each having a duration *t/n* and decoding the second step of the second code word in *n* sequential sub-steps each having a duration *t/n* and decoding the first step of the third code word in *n* sequential sub-steps each having a duration *t/n*; and~~

~~while receiving the fifth code word, decoding the fourth step of the first code word in *n* sequential sub-steps each having a duration *t/n* and decoding the third step of the second code word in *n* sequential sub-steps each having a duration *t/n* and decoding the second step of the third code word in *n* sequential sub-steps each having a duration *t/n* and decoding the first step of the fourth code word in *n* sequential sub-steps each having a duration *t/n*.~~

28. (currently amended) A method of operating a computing system with a Reed-Solomon decoding application comprising the steps:

sequentially receiving *m* portions of the Reed-Solomon-encoded string of data in *T* seconds, the string of data including *n* symbols, *n/m* equaling a first integer that is greater than one;

sequentially calculating *m* respective partial syndromes for the *m* portions of the string, the processor circuit operable to calculate each of the *m* partial syndromes in *T/m* seconds;

from the *m* partial syndromes, sequentially calculating the coefficients of *m* respective sets of error locator polynomials, the processor circuit operable to calculate each set of coefficients in *T/m* seconds;

sequentially determining *m* respective sets of roots for the sets of the error locator polynomials, the processor circuit operable to determine each set of roots in *T/m* seconds;

for each of the *m* sets of roots, sequentially determining the magnitude of a respective error in *T/m* seconds and correcting each of the errors in *T/m* seconds. ~~1. receive the Reed-Solomon-encoded string of data in *t* seconds;~~

~~2. calculate a syndrome in said *t* seconds;~~

~~3. calculate the coefficients of an error-locator polynomial in said *t* seconds;~~

~~4. determine the roots of an error-locator polynomial in said *t* seconds; and~~

~~5. determine the magnitude of an error and correcting the error in said *t* seconds.~~

29 – 30. Canceled.

31. (original) The method of claim 28 wherein the computing system executes a Berlekamp-Massey Algorithm to calculate the coefficients of the error locator polynomial.

32. (original) The method of claim 28 wherein -the processor circuit executes a Chien Search to determine the roots of the error locator polynomial.

33. (original) The method of claim 28 wherein the processor circuit executes a Forney Algorithm to determine the magnitude of the errors in the received digital code word.

34 – 36. Canceled.

37. (new) The method of claim 27 wherein the portions of the five Reed-Solomon code words each have a same size.